Automated Analysis of a I.argc-Scale Sky Survey: The SKI CAT Systcm

Usama M. Fayyad
Al Group MIS 525-3660
Jet Propulsion Laboratory
California Institute of ~'ethnology
Pasadena, CA 9] 109
Fayyad@aig.jpl.nasa.gov

Nicholas Weir anti S. G. Djorgovski
Department of Astronomy
M/S 105-24
California Institute of Technology
Pasadena, CA 91125
{weir,george} @deimos.caltech.edu

ABSTRACT

We describe the application of decision tree based classification techniques to the development of an automated tool for the reduction of a large scientific data set. The 2nd Palomar Observatory Sky Survey (1'0SS-11) provides comprehensive coverage of the northern celestial hemisphere in the form of digitized photographic plates whose quality will probably not be surpassed in the next ten to twenty years. The images arc expected to contain on the order of 107 galaxies and 10^8 stars. Astronomers wish to determine which of these sky objects belong to various classes of galaxies and stars. The size of this data set precludes manual analysis. Our approach is to develop a software system which integrates the functions of independently developed techniques for image processing, data classification, and database management. Public domain image processing routines arc used to identify sky objects and to extract a set of basic features for each object. These features arc used to select a useful and robust set of attributes for classifying sky objects. The GID3* anti O-BTree decision tree learning algorithms, in conjunction with the RULER system for statistically pruning and merging multiple trees, are used to classify the detected objects. The results indicate that our approach is well-suited to the problem. Using higher resolution image sources covering minute portions of the survey, the learning algorithms produced classifiers that can classify objects in the survey that arc too faint for visual classification with an accuracy level of about 94%. Not only does this increase the number of objects in the final catalog by three-fold (75% of objects in an image are faint), it allows us to catalog classified sky objects that arc at least onc magnitude fainter than objects classified in sky surveys to date. SKICAT represents a system in which machine learning played a powerful and enabling role, and solved a difficult, scientifically significant problem. The primary benefits of the approach are increased data reduction throughput, repeatability, and consistency of classification.

Keywords: Machine Learning Application (large-scale), decision tree learning, Scientific Data Analysis, Image Databases.

1. INTRODUCTION

in this paper we present an application of machine lemming techniques to the automation of the task of cataloging sky objects in digitized sky images. The Sky Image Classification and Archiving Tool (SKICAT) is being developed for use on the images resulting from the 2nd Palomar Observatory Sky Survey (POSS-II) conducted by the California Institute of Technology (Caltech). The photographic plates collected from the survey are being digitized at the Space Telescope Science Institute (STScI). This process will result in about 3,000 digital images of roughly $23{,}000^2$ pixels[1] each. The survey consists of over 3 terabytes of data containing on the order of 10^7 galaxies, 10^8 stars, and 10^5 quasars.

The first step in analyzing the results of a sky survey is to identify, measure, and catalog the detected objects in the image into their respective classes. Once the objects have been classified, further scientific analysis can proceed. For example, the resulting catalog may be used to test models of the formation of large-smle structure in the universe, probe galactic structure from star counts, perform automatic identifications of radio or infrared sources, and so forth. The task of reducing the images to catalog entries is a laborious time-consuming process. A manual approach to constructing the catalog implies that many scientists need to expend large amounts of time on a visually intensive task that may involve significant subjective judgment. The goal of our project is to automate the process, thus alleviating the burden of cataloging objects from the scientist and providing a more objective methodology for reducing the data sets. Another goal of this work is to classify objects whose intensity (isophotal magnitude) is too faint for recognition by inspection, hence requiring an automated classification procedure. Faint objects constitute the majority of objects on any given plate. We target the classification of objects that are at least one magnitude fainter than objects classified in previous surveys using comparable photographic material.

The goals of this paper are:

1. to introduce the machine learning techniques we used and compare their performance to other' alternatives such as neural networks,
2. to give a general, high-level description 01 the current application domain.
3. to report on the successful results which exceeded our initial goals for this problem.

We therefore do not provide. the details of either the lemming algorithms or the technical aspects of the domain. We aim to point out an instance where the learning algorithms proved to a be useful and powerful tool in the automation of scientific data analysis.

2. MACHINE LEARNING BACKGROUND

The growing number of large diagnostic and scientific databases provides an important niche for machine learning techniques. A database that stores instances of diagnostic tasks is typically accessed by keyword or condition look up. As the size of the database grows, such an approach becomes ineffective since a query may easily return hundreds of matches making simple case-based usage impractical. For large scientific databases the problem is to search for and detect patterns of interest, or to perform pre-processing necessary for subsequent analysis. Sizes are now becoming too large for manual processing. Learning techniques can Serve as effective tools for aiding in the analysis, reduction, and visualization of large scientific databases.

2.1. INDUCTION OF DECISION TREES

A particularly efficient method for extracting rules from data is to generate a decision tree [Brei84, Quin86]. A decision tree consists of nodes that are tests on the attributes. The outgoing branches of a node correspond to all the possible outcomes of the test at the node. The examples at a node in the tree are thus *partitioned* along the branches and each child node gets its corresponding subset of examples. A popular algorithm for generating derision trees is Quinlan's ID3 [Quin86] with extended versions called C4 [Quin90].

[1] Each pixel consists of 16 bits of data representing intensity in one of three colors.

ID3 starts by placing all the training examples at the root node of the tree. An attribute is selected to partition the data. For each value of the attribute, a branch is created and the corresponding subset of examples that have the attribute value specified by the branch are moved to the newly created child node. The algorithm is applied recursively to each child node until either all examples at a node are of one class, or all the examples at that node have the same values for all the attributes. Every leaf in the decision tree represents a classification rule.

Note that the critical decision in such a top-down decision tree generation algorithm is the choice of attribute at a node. The attribute selection in ID3 and C4 is based on minimizing an information entropy measure applied to the examples at a node. The measure favors attributes that result in partitioning the data into subsets that have low class entropy. A subset of data has low class entropy when the majority of examples in it belong to a single class. The algorithm basically chooses the attribute that provides the locally maximum degree of discrimination between classes. For a detailed discussion of the information entropy selection criterion see [Quin86, Fayy91].

2.2. THE GID3* AND O-BTREE ALGORITHMS

The criterion for choosing the attribute clearly determines whether a "good" or "bad" tree is generated by the algorithm[2]. Since making the optimal attribute choice is computationally infeasible, ID3 utilizes a heuristic criterion which favors the attribute that results in the partition having the least information entropy with respect to the classes. This is generally a good criterion and often results in relatively good choices. However, there are weaknesses inherent in the ID3 algorithm that are due mainly to the fact that it creates a branch for each value of the attribute chosen for branching. The overbranching problem in ID3 leads to several probelms, since in general it may be the case that only a subset of values of an attribute are of relevance to the classification task while the rest of the values may not have any special predictive value for the classes. These extra branches are harmful in three ways [bayy91 ,Fayy93a]:

1. They result in rules that are overspecialized. The leaf nodes that are the descendants of the nodes created by the extraneous branches will be conditioned on particular irrelevant attribute values.
2. They unnecessarily partition the data, thus reducing the number of examples at each child node. The subsequent attribute choices made at such child nodes will be based on an unjustifiably reduced subset of data. The quality of such choices is thus unnecessarily reduced.
3. They increase the likelihood of occurrence of the missing branches problem. This problem occurs because not every possible combination of attribute values is present in the examples (see [I:ayy91, Fayy93a] for more details).

The GID3* algorithm was designed mainly to overcome this problem. It utilizes a vector distance measure applied to the class vectors of an example partition, in conjunction with the entropy measure, to create for each attribute a *phantom attribute* that has only a subset of the original attribute's values. We generalized the ID3 algorithm so that it does not necessarily branch on each value of the chosen attribute, GID3* can branch on arbitrary individual values of an attribute and "lump" the rest of the values in a single *default branch*. Unlike the other branches of the tree which represent a single value, the default branch represents a subset Of values Of an attribute. Unnecessary subdivision of the data may thus be reduced. See [Fayy91,Fayy93a] for more details and for empirical evidence of improvement.

The O-Btree algorithm [Fayy92b] was designed to overcome problems with the information entropy selection measure itself. O-Btree creates strictly binary trees and utilizes a measure from a family of measures (C-SEP) that detects class separation rather than class impurity. Information entropy is a member of the class of impurity measures. O-Btree employs an orthogonality measure rather than entropy for branching. For details on problem with entropy measures and empirical evaluation of O-Btree, the reader is referred to [Fayy91,Fayy92b].

Both O-Btree and GID3* differ from ID3 and C4 along one additional aspect: the discretization algorithm used at each node to discretize continuous-valued attributes. Whereas ID3 and C4 utilize a binary interval discretization algorithm, we utilize a generalized version of that algorithm which

[2] See [Fayy90,Fayy91] for the details of what we formally mean by one decision tree being better than another

derives multiple intervals rather than strictly two. For details and empirical tests showing that this algorithm does indeed produce better trees see [Fayy91,Fayy93b]. We have found that this ability dots improve performance considerably in several domains.

2.3. THE RULER SYSTEM

'1'here arc limitations to decision tree generation algorithms that derive from the inherent fact that the classification rules they produce originate from a single tree. This fact was recognized by practitioners early on [Brei94,Quin87]. Tree pruning is used to overcome the fact that in any good tree there arc always leaves that arc overspecialized or predict the wrong class. The very reason which makes decision tree generation efficient: the fact that data is quickly partitioned into ever smaller subsets, is also the reason why overspecialization or incorrect classification occurs. It is our philosophy that once we have good, efficient, decision tree generators, they could be used to generate multiple trees and only the best rules in each tree arc kept. We initially developed the RIST system [Chen90] which later evolved into the RULER system to implement such a scheme. Figure 1 gives an overview of the RULER system.

RULER starts with a data set, and randomly divides it into a training subset and test subset. A decision tree is generated from the training set and its rules arc tested on the corresponding test set. Using Fisher's exact test [Finn63] (the exact hypergeometric (distribution) RULER evaluates each condition in a given rule's preconditions for relevance to the class predicted by the rule. It computes the probability that the condition is correlated with the class by chance [3]. If this probability is higher than a small threshold (say 0.01), the condition is deemed irrelevant and is pruned. In addition, RULER also measures the merit of the entire rule by applying the test to the entire precondition as a unit. This process serves as a filter which passes only robust, general, and correct rules.

By gathering a large number of rules through iterating on randomly subsampled training sets, RULER builds a large rule base of robust rules that collectively cover the entire original data set of

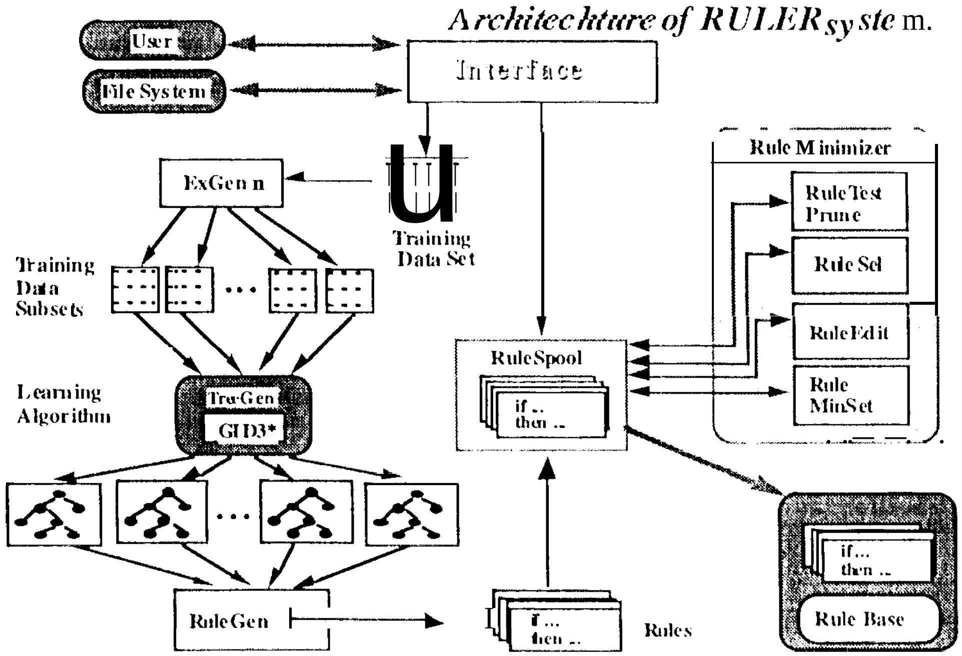

Figure 1. Architecture of the RULER Rule Induction System

[3] The Chi-square test is actually an approximation to Fisher's exact test when the number of test examples is large. We use Fisher's exact test because it is robust for small and large data sets.

examples, A greedy covering algorithm is then employed to select a minimal subset of rules that covers the examples. The set is minimal in the sense that no rule could be removed without losing complete coverage of the original training set.

Using this method, wc can typically produce a robust set of rules that has fewer rules than any of the original decision trees that were used to create it. Furthermore, any learning algorithm that produces rules can be used as the rule generating component. We usc decision tree algorithms because they constitute a fast and efficient method for generating a set of rules from a training set. This allows us to iterate many times without requiring extensive amounts of time and computation.

Now that wc have covered all the relevant components of the learning system, wc shall turn our attention to the task of automating sky object classification.

3. CLASSIFYING SKY OBJECTS

Due to the large amounts of data being collected, a manual approach to classifying sky objects in the images is infeasible (it would require on the order of tens of man years). Existing computational methods for processing the images will preclude the identification of the majority of objects in each image since they arc at levels too faint for traditional recognition algorithms or even manual inspection/analysis approaches. Our main objective is to provide an effective, objective, and examinable basis for classifying sky objects.

The photographic plates collected from the survey are being digitized at the Space Telescope Science Institute (STScI). This process will result in about 3,000 digital images of roughly 23,0002 pixels each. Low-level image processing and object separation is performed by the FOCAS image processing software developed at Bell Labs [Jarv81, Vald82]. In addition to defining the objects in each image, FOCAS also produces basic attributes describing each object. A digitized plate is subdivided into a set of partially overlapping frames. Each frame represents a small part of the plate that is small enough to be manipulated and processed conveniently, Figure 2 depicts the ovc.rail architecture of the SKICAT System. The discussion below will explain the loop in the bottom left-hand comer in which machine learning is employed in the attribute measurement process. The image processing steps that a digitized plate goes through are:

1. Select a frame from the digitized plate.
2. Detection: detect contiguous pixels in the image that are to be grouped as one object (standard image processing).
3. Perform more accurate local sky determination for each detected object.
4. Evaluate parameters for each object independently: we initially measured 18 b~{sc-level *attributes*.
5. Split objects that arc "blended" together and re-evaluate attributes.
6. **AUTOPSF**: select a subset of the objects in the frame and designate them as being "sure-thing" stars, form PSF template (see below).
7. Measure *resolution scale* and *resolution fraction* attributes for each object: These arc obtained by fitting the object to the PSF template of sure-thing stars formed in step 6.
8. Measure additional normalized attributes (bringing total attributes to 40)
9. Classify objects in image.

All steps arc automated except for steps 6 and 9. Step 6 needs further elaboration. The goal of this step is to define the two resolution attributes mentioned in step 7. These attributes arc parameters of a template defined on a point spread function (PSF). The template is computed over a subset of objects identified as sure-thing stars, The sure-thing stars arc selected by the astronomer. They represent the "archetypal" stars in that image, Once the stars arc selected, the template fitting and resolution parameter measurements arc computed automatically. Thus in order to automate steps 1-8 wc need to automate the star selection step (6). We refer to this problem as the *star selection subproblem*.

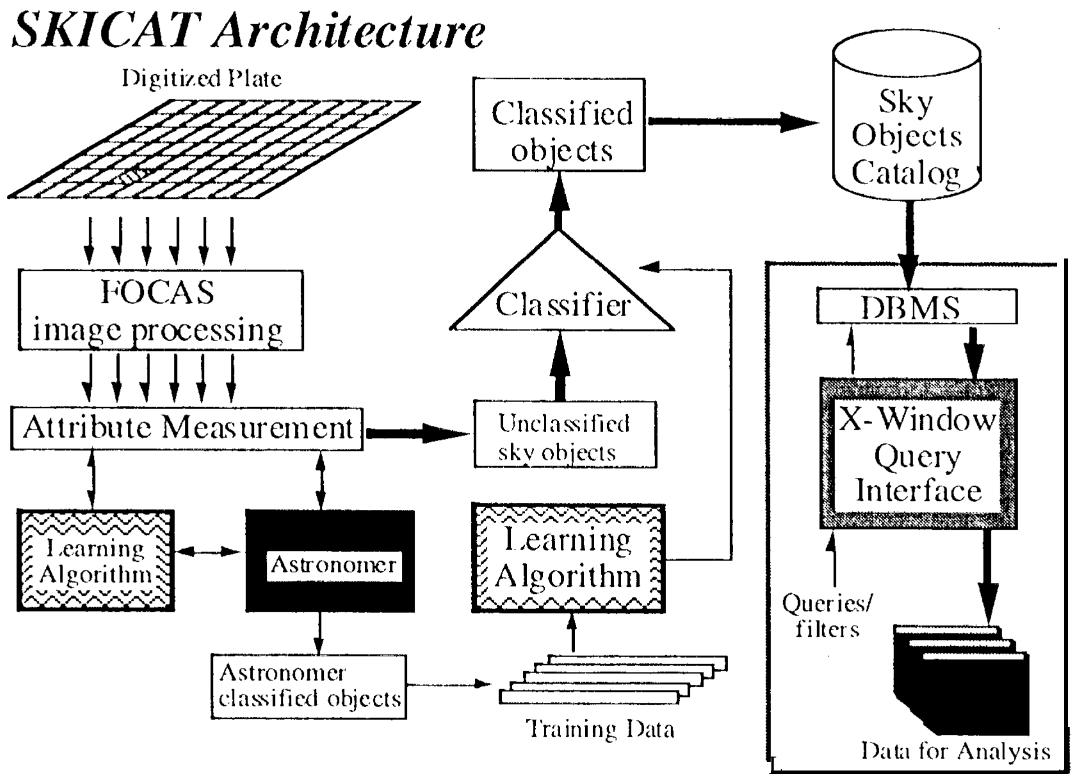

Figure 2. Architecture of the SKI CAT System

Thirty base-level attributes are measured in step 4. These are generic quantities used typically used in astronomical analyses [Vald82]. A subset of these is listed below.

- isophotal magnitude
- isophotal area
- core magnitude
- core luminosity
- sky brightness
- sky sigma (variance)
- image moments (8): ir1, ir2, ir4, r1, r2, ixx, iyy, anti ixy.
- eccentricity (ellipticity)
- orientation
- semi-major axis
- semi-minor axis.

Once all attributes, including the resolution attributes, for each object are measured, step 8 involves performing the final classification for the purposes of the catalog. We are currently classifying objects into four major categories: star (s), star with fuzz (**sf**), galaxy **(g)**, and artifact **(long).** We may later refine the classification into more classes, however, classification into one of the four classes represents our initial goal.

3.1. Classifying Faint Objects and the Use of CCD Images

in addition to the scanned photographic plate, we have access to CCD images that span several small regions in some of the frames. CCD images are obtained from a separate telescope. The main advantage of a CCD image is higher resolution and signs]-to-noise ratio at fainter levels. Hence, many of the objects that are too faint to be classified by inspection of a photographic plate, are easily classifiable in a CCD image. in addition to using these images for photometric calibration of the photographic plates, we make use of CCD images in two very important ways for the machine learning aspect:

1. CCD images enable us to obtain class labels for faint objects in the photographic plates.
2. CCD images provide us with the means to reliably evaluate the accuracy of the classifiers obtained from the decision tree learning algorithms,

Recall that the image processing package FOCAS provides the measurements for the base-level attributes (and the resolution attributes after star selection) for each object in the image. in order to produce a classifier that classifies faint objects correctly, the learning algorithm needs training data consisting of faint objects labeled with the appropriate class. The class label is therefore obtained by examining the CCD frames. Once trained on properly labeled objects, the learning algorithm produces a classifier that is capable of properly classifying objects based on the values of the attributes provided by FOCAS. Hence, in principle. the classifier will be able to classify objects in the. photographic image that arc simply too faint for an astronomer to classify by inspection. Using the class labels, the learning algorithms are basically being used to solve the more difficult problem of separating the classes in the multi-dimensional space defined by the set of attributes derived via image processing. This method is expected to allow us to classify objects that arc at least one magnitude fainter than objects classified in photographic sky surveys to date.

3,2. RESULTS FOR THE CLASSIFICATION PROBLEM

Starting with digitized frames obtained from a single digitized plate, we performed initial tests to evaluate the accuracy of the classifiers produced by the machine learning algorithms ID3, GID3*, and O-13 Tree. The data consisted of objects collected from four different plates from regions for which we had CCD image coverage (since this is data for which true accurate classifications are available). The learning algorithms arc trained on a data set from 3 plates and tested on data from the remaining plate for cross validation. This estimates our accuracy in classifying objects across plates. Note that the plates cover different regions of the sky and that CCD frames cover multiple minute portions of each plate. The training data consisted of 1,688 objects that were classified manually by one of the authors (NW) by examining the corresponding CCD frames. It is noteworthy that for the majority of these objects, the astronomer would not be able to determine the classes by examining the corresponding survey (digit ized photographic) images. All attributes used by the learning algorithms arc derived from the survey images and not from the higher resolution CCD flames.

Using all the attributes derived in step (8) including the two resolution attributes derived in step ('7), the classification results are shown in Table 1.

11)3		GID3*		O-Btree		RULER	
#l ules	accuracy	#rules	accuracy	#rules	accuracy	#rules	accuracy
73	75.6%	58	90.1940	54	91.296	45	94.2%

'1'able 1. Summary of results using all attributes.

The results for RULER above are shown for using O-Btree as the decision tree generation component and were obtained by cycling through tree generation and rule merging 10 times. Using ID3, the results were not as good: the accuracy in this case was only around 85%. Results with using GID3* as the tree generating component for ruler are similar to O-Btree's.

However, when the same experiments were conducted without using the *resolution scale* and *resolution fraction* attributes of step 6, the results were significant! y worse. The error rates jumped above 20% for O-B"J'ret, above 25% for GID3*, and above 30% for ID3. The respective sizes of the trees grew significantly as well.

The initial results may be summarized as follows:

1. Algorithms GID3* and O-BTree produced significantly better trees than ID3.
2. [classification accuracy results of better than 90% were obtained when using two user-defined attributes: *resolution fraction* and *resolution scale*.

3. Classification results were not as reliable and stable if we exclude the two resolution attributes.

We took this as evidence that the resolution attributes are very important for the classification task. Hence wc turned to addressing the star selection subproblem in order to automate step 6 above. Furthermore, the results point out that the GID3* and O-BTree learning algorithms arc more appropriate than ID3 for the final classification task. As expected, the use of RULER resulted in improvement in performance.

3.3. AUTOMATING THE STAR SELECTION PROCESS

Based on the initial results of the previous section, it was determined that using the resolution attributes is necessary since without them the error rates were significantly worse. Wc do not have the option of leaving star selection as a manual step in the process, since it is a time consuming task and will easily become the bottleneck in the system. We decided to usc a machine learning approach to solve the star selection subproblem.

The star selection subproblem is a binary classification problem. Given a set of objects in an image, the goal is to classify them as sure-thing stars and non-sure-thing stars. Unlike the overall classification problem, the star selection problem turned out to bc a much easier classification problem. The data objects from all three plates described above were classified manually by one of the authors (NW.) into *sure-slars, non-sure-stars*, and *unknowns*. The goal of the learning subproblem is to construct classifiers for selecting out sure-stars from any collection of sky objects. The results of applying the learning algorithms to the data sets described above, using only the attributes dc.rived in step 5 of course, gave the results shown in Table 2.

ID3		GID3*		O-Btree	
#rules	accuracy	#rules	accuracy	#rules	accuracy
41	95%	35	97.3%	29	98.7%

Table 2. Summary **of results using all attributes.**

In this case, using RULER with O-Btree did not change the results significantly. Note that a 98.770 accuracy rate on this subproblem is more than sufficient to indicate that this subproblem is essentially completely solved. Consequently, this allows us to automate all the steps in the plate processing and obtain an overall classification rate of better than 94% as shown in "l'able 1. One note about this learning subproblem: the results reflect the accuracy in selecting sure-thing stars and not the classification error rate. in other words, we only care about the performance in terms of sure-thing stars selected correctly. Sure-stars classified as galaxies or unknowns dots not concern us since all wc need is a subset of good stars to fit the PSF template to. Since this is not the main classification task, wc only present the relevant performance aspects to avoid confusion.

3.4. CROSS-PLATE ROBUSTNESS & COMPARISON WITH NEURAL NETS

In order to achieve stable classification accuracy results on classifying data from different plates, wc had to spend some effort in defining some normalized attributes that arc less sensitive to plate-to-plate variation. It was determined that the base-level attributes such as area, background-sky-levels, and average intensity arc image-dependent as wc1l as object-dependent. It was also determined that a new set of user-defined attributes needed to be formulated. These attributes were to be computed automatically from the data, anti arc defined such that their values would bc normalized across images and plates. A typical technique wc used to derive such attributes is to derive non-linear curves in two dimensions defined by two of the base-level attributes and then define a new attribute to be the distance of each object in the 2-D plane to that curve. These quantities arc ones that astronomers usc, and many of them have physical interpretations.

It is beyond the scope of this paper to give the detailed (definitions of these new attributes. As expected, defining the new "normalized" attributes raised our performance on both intro- and inter-plate classification to acceptable levels varying between 92% and 98% accuracy with an average of **94%,** Note that without these derived attributes the cmss-plate classification accuracy drops to 60%-80% levels when classifying data from different plates. Our encoding of the.sc attributes represents an implicit imparting of more domain knowledge to the learning algorithm.

In order to compare against other learning algorithms, and to preclude the possibility that a decision tree based approach is imposing *a priori* limitations on the achievable classification levels, wc tested several neural network algorithms for comparison. The results indicate that neural network algorithms achieve similar, and sometimes worse performance than the decision trees. The neural net learning algorithms tested were:

1. traditional backpropagation,
2. conjugate gradient optimization, and
3. variable metric optimization.

Unlike backpropagation, the latter two are training algorithms work in batch mode and use standard numerical optimization techniques in clinging the network weights [Hert91]. They compute the weight adjustments simultaneously using matrix operations based on the total error of the network on the entire 'training set. Their main advantage over traditional backpropagation is the significant speed-up in training time.

The results can be summarized as follows: The performance of the neural networks was fairly unstable and produced accuracy levels varying between 30% (no convergence) and 95%. The most common range of accuracy on average was between 76% and 84%. Note that we had to perform multiple trials, each time varying:

1. the number of internal nodes in the (single) hidden layer,
2. the initial weight settings for a given network architecture, and
3. the learning rate constant for backpropagation.

Upon examining the results of the empirical evaluation, wc concluded that the neural net approach did not offer any clear advantages over the decision tree based learning algorithm. Although neural networks, with extensive training and several training restarts with different initial weights to avoid local minima, could match the performance of the decision tree classifier, the decision tree approach still holds several major advantages. The most important is that the tree is easy for domain experts to understand. In addition, unlike neural network learning algorithms, the decision tree learning algorithms GID3* and O-BTree do not require the specification of parameters such as the size Of the neural net, the number Of hidden layers, and random trials with different initial weight settings. Also, the required training time is orders of magnitude faster than the training time required for a neural network approach.

The stability of the performance of the decision tree algorithms, and the fact that a decision tree (or classification rule) is a lot easier to interpret and understand than a neural network, wc decided to adopt the decision tree approach in this domain.

3.5. VERIFICATION AND RELIABILITY ESTIMATES

As mentioned earlier, in addition to using the CCD frames to derive training data for the machine learning algorithms, wc also use them to verify and estimate the performance of our classification technique. This is done by testing on data sets that are drawn independently from the training data. An additional source of internal consistency checks comes from the fact that the plates, and the frames within each plate are partially overlapping. Hence, objects inside the overlapping regions will be classified in more than one context. By measuring the rate of conflicting classifications, we can obtain further estimates of the statistical confidence in the accuracy of our classifier. For the purposes of the final catalog production, a method is being designed for resolving conflicts on objects within regions of overlap. We have not yet collected reportable results on this aspect of the problem.

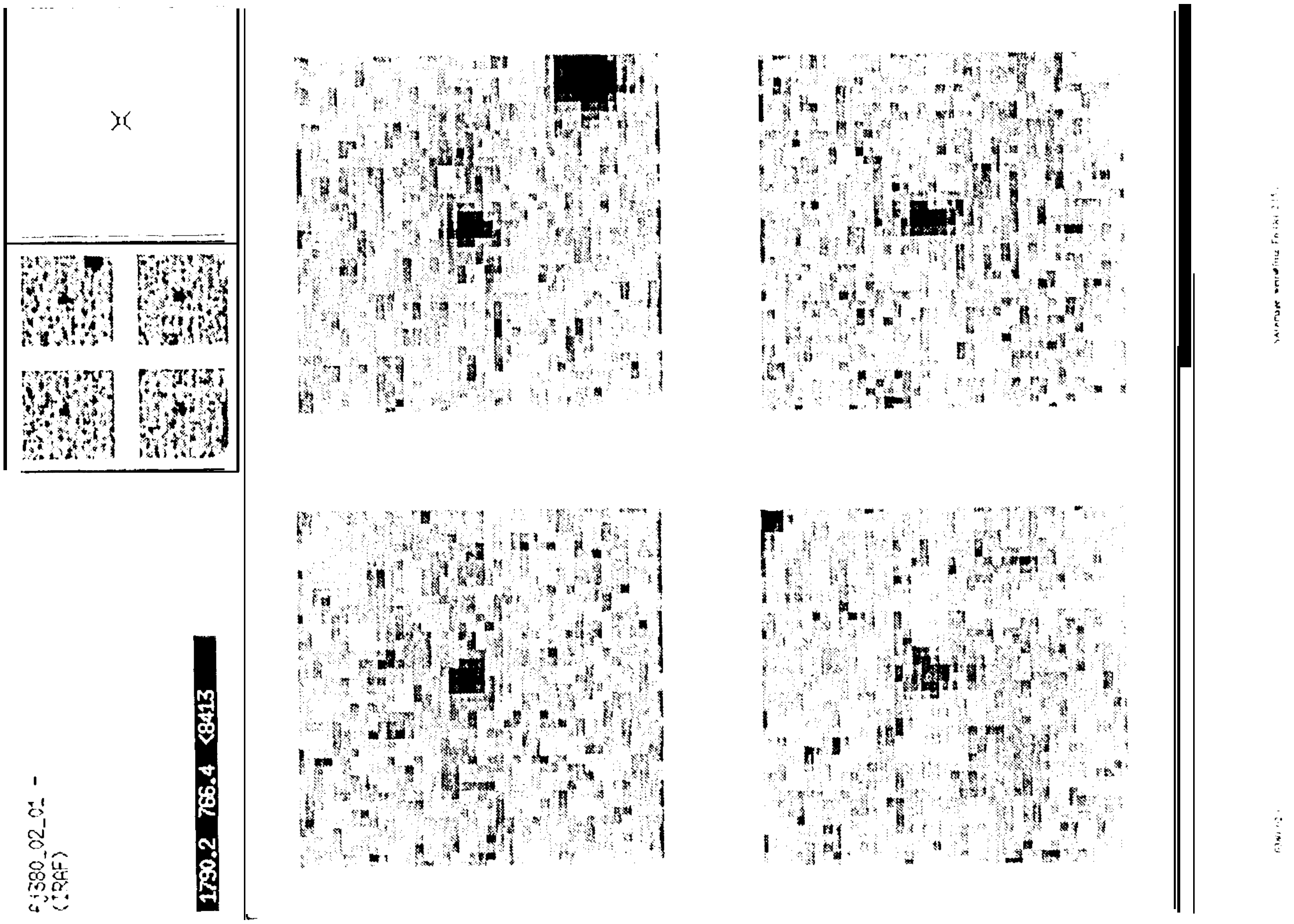

Figure 3. An illustrative example: four faint sky objects.

In order to demonstrate the difficulty and significance of the classification results presented so far, consider the example shown in Figure 3. This figure shows four image patches each centered about an object that was classified by SKICAT as a galaxy. These images were obtained from a plate that was not provided to SKICAT in the training cycle, According to several astronomers, the objects shown in Figure 3 do not appear to be galaxies. As a matter Of fact, an astronomer visually inspecting these images would be hard pressed to decide whether they are stars or galaxies. However, in every one of these cases, upon retrieving the corresponding higher resolution CCD images of these objects, it was very clear that they were indeed galaxies. Note that SKICAT produced the prediction based on the lower resolution survey images (shown in the figure). This example illustrates how the SKICAT classifier can correctly classify the majority of faint objects which even the astronomers cannot classify. Indeed, the results indicate that SKICAT performs this task with an accuracy better than 91% (for the faintest objects in the survey).

4. CONCLUSIONS AND FUTURE WORK

In this paper, we gave a brief overview' of the machine learning techniques we used for automating the sky object cataloging problem. The SKICAT system is expected to speed up catalog generation by one to two orders of magnitude over traditional manual approaches to cataloging. This should significantly reduce the cost of cataloguing survey images by the equivalent of tens of astronomer mm-years. in addition, SKICAT classifies objects that are at least one magnitude fainter than objects cataloged in previous surveys. Finally, this project represents a step towards the development of an objective, reliable automated sky object classification method.

The initial results of our effort to automate sky object classification in order to automatically reduce the images produced by POSS-II to sky catalogs are indeed very encouraging. We have exceeded our initial accuracy target of 90%. This level of accuracy is required for the data to be useful in testing or refuting theories on the formation of large structure in the universe and on other phenomena of interest to astronomers,

In addition to using machine learning techniques to automate classification, we used them to aid in the attribute measurement process. Since measurement of the resolution attributes requires interaction with the user in selecting sure-thing stars for template fitting, we used the same machine learning approach to automate the star selection process. By defining additional "normalized" image-independent attributes, we were able to obtain high accuracy classifiers for star selection within and across photographic plates. This in turn allows us to automate the computation of the powerful resolution attributes for each object in an image.

Final object classification will be, to some extent, also a matter of scientific choice. While objects in every catalog will contain a classification entry, all of the object attribute.s will be recorded as well. One could therefore reclassify any portion of the survey using alternative criteria better suited to a particular scientific goal (e.g. star catalogs vs. galaxy catalogs). The catalogs will also accommodate additional attribute entries, in the event other pixel-based measurements are deemed necessary. An important feature of the survey analysis system will be to facilitate such detailed interactions with the catalogs. The catalog generated by SKICAT will eventually contain about a billion entries representing hundreds of millions of sky object. Unlike the traditional notion of a static printed catalog, we view our effort as targeting the development of a new generation of scientific analysis tools that render it possible to have a constantly evolving anti growing catalog. Without the availability of these tools for the first survey (POSS-I) conducted over 4 decades ago, only a small percentage of the data was used and only specific areas of interest were studied. In contrast, we are targeting a comprehensive sky catalog that will be available on-line for the use of the scientific community.

As part of our plans for the future we plan to begin investigation of the applicability of unsupervised learning (clustering) techniques such as AUTOCLASS [Chee88] to the problem of discovering clusters or groupings of interesting objects. The initial goals will be to answer the following two questions:

1. Are the classes of sky objects used currently by astronomers justified by the data: do they naturally arise in the data?
2. Are there other classes of objects that astronomers were not aware of because of the difficulty of dealing with high dimensional spaces defined by the various attributes? Essentially this is a discovery problem.

The longer term goal is to evaluate the utility of unsupervised learning techniques as an aid for the types of analyses astronomers conduct after objects have been classified into known classes. Typically, astronomers examine the various distributions of different types of objects to test existing models of the formation of large-scale structure. in the universe. Armed with prior knowledge about properties of interesting clusters of sky objects, a clustering system can search through catalog entries and point out potentially interesting object clusters to astronomers. This will help astronomers catch important patterns in the data that may otherwise go unnoticed due to the sheer size of the data volumes.

ACKNOWLEDGMENTS

We would like to thank the Sky Survey team for their expertise and effort in acquiring the plate material. The POSS-II is funded by grants from the Eastman Kodak Company, The National Geographic Society, The Samuel Oschin Foundation, NSF Grants AST 84-08225 and AST 87-19465, and NASA Grants NGL 05002140 and NAGW 1710. We thank Joe Roden for help on evaluating the performance of the learning algorithms. This work was supported in part by a NSF graduate fellowship (N. Weir), the Caltech President's Fund, NASA contract NAS5-31348 (S. Djorgovski and N. Weir), and the NSF PYI Award AST-9 157412 (S. Djorgovski).

The work described in this paper was carried out in part by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

REFERENCES

[Brei84] Breiman, L., Friedman, J.}]., Olshen, R. A., and Stone, C.J. (1984). Classification and Regression Trees. Monterey, CA: Wadsworth & Brooks,

[Chee88] Cheeseman, P., Self, M., Kelly, J., Taylor, W. Freeman, D. and Stutz, J. (1988) "Bayesian classification. " Proceedings of the Seventh National Conference on Artificial Intelligence AAAI-88, pp. 607-6611, Saint Paul, MN.

[Chen90] Cheng, J., Fayyad, U.M., Irani, K.B. and Qian, Z. (1 990) "Applications of machine learning techniques in semiconductor manufacturing." Proceedings of the SPIE Conference on Applications of Artificial Intelligence VII I. pp. 956-965, Orlando, Fl.

[Feig81] Feigenbaum, E.A. (1981) "Expert systems in the 1980s. " in Bond, A. (Ed.) State of The Art Report on Machine Intelligence. Maidenhead: Pergamon-Infotech.

[Fayy90] Fayyad, U.M. and Irani, K. B. (1990). "What should be minimized in a decision tree?" Proceedings of Eighth National Conference on Artificial Intelligence AAAI-90, Boston, MA.

[Fayy91] Fayyad, U.M. (1991). On the induction of Decision Trees for Multiple Concept Learning. PhD Dissertation, EECS Dept. The Universit y of Michigan.

[Fayy92a] Fayyad, U.M. and Irani, K.B. (1 992) "On the handling of continuous-valued attributes in decision tree generation. " Machine Learning, vol.8, no.2.

[Fayy92b] Fayyad, U.M. and Irani, K.B. (1992) "The attribute selection problem in decision tree generation. " Proceecdings of the Tenth National Conference on Artificial Intelligence, AAA1-92. San Jose, CA.

[Fayy93a] Fayyad, U.M. (submitted) "Attribute value sub-branching in decision tree generation. " submitted to Eleventh National Conference on Artificial Intelligence, AA AI-93. San Jose, CA.

[Fayy93b] Fayyad, U.M. (submitted) "MLdti-interval discretization of continuous-valued attributes for classification learning. " submitted to Tenth International Conference on Machine Learning. U.Mass., Amherst.

[Finn63] Finney, D. J., Latscha, R., Bennett, B. M., and Hsu, P. (1963). Tables for Testing Significance in a 2x2 Contingency '1'able. Cambridge: Cambridge University Press.

[Hert91] J. Hertz, A. Krogh, and R.G. Palmer (1991) introduction to the Theory of Neural computation. Reading, MA: Addison-Wesley.

[Jarv81] Jarvis, J. and Tyson, A. (1981) Astronomical Journal 86:41.

[Quin86] Quinlan, J.R. (1986) "The induction of decision trees. " Machine l earning vol. 1, no. 1.

[Quin90] Quinlan, J.R. (1990) "Probabilistic decision trees. " Machine Learning: Arr Artificial Intelligence Approach vol. 111. Y. Kodratoff & R. Michalski (eds.) San Mateo, CA: Morgan Kaufmann.

[Vald82] Valdes (1982) Instrumentation in Astronomy IV, SPIE vol. 331, no. 465.